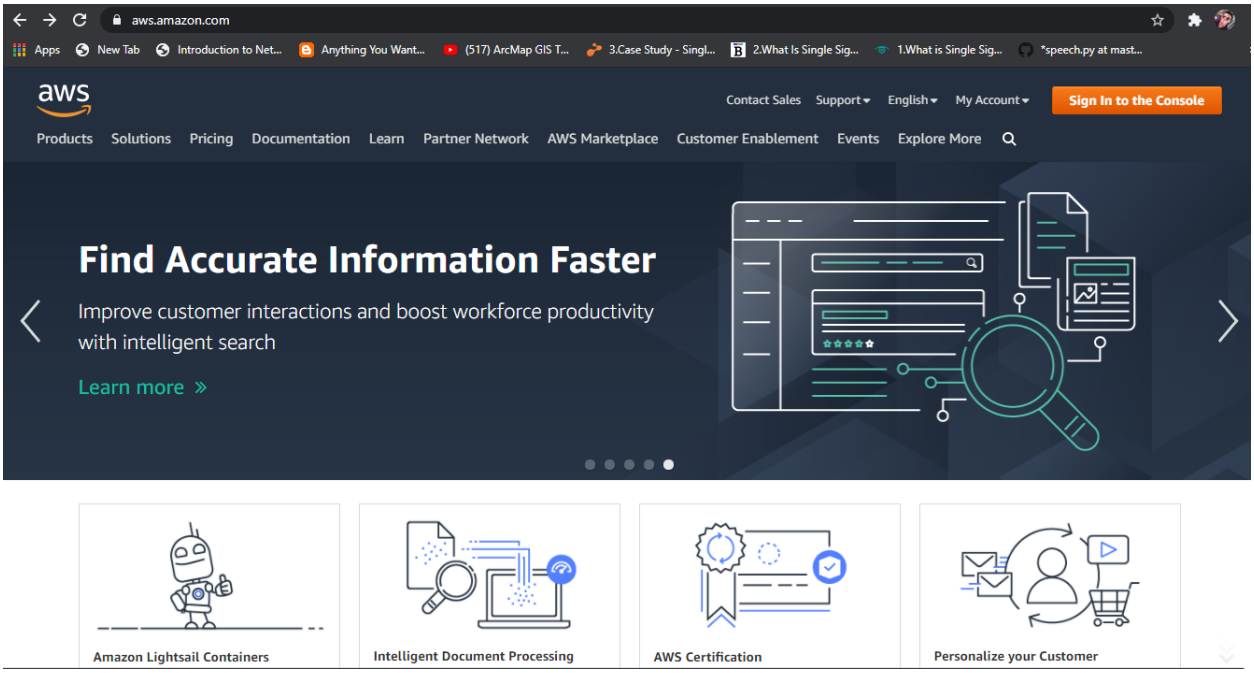
DOCUMENTATION

London Stock Exchange Group Sri Lanka - Exchange House

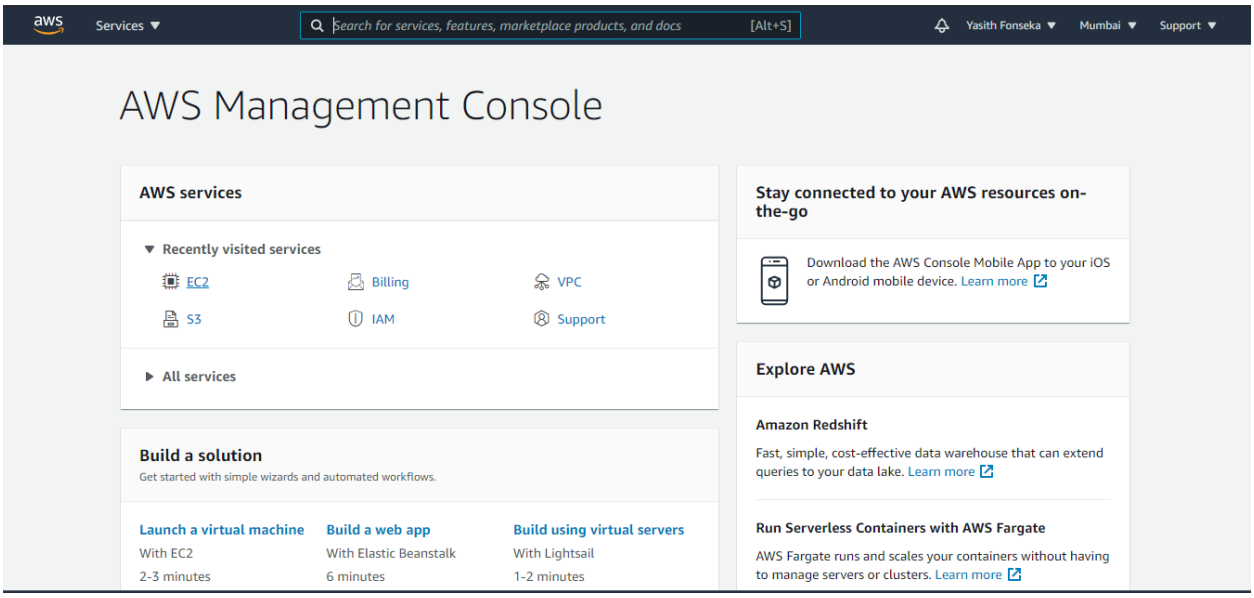
AYS FONSEKA

PART 01 – Scripting

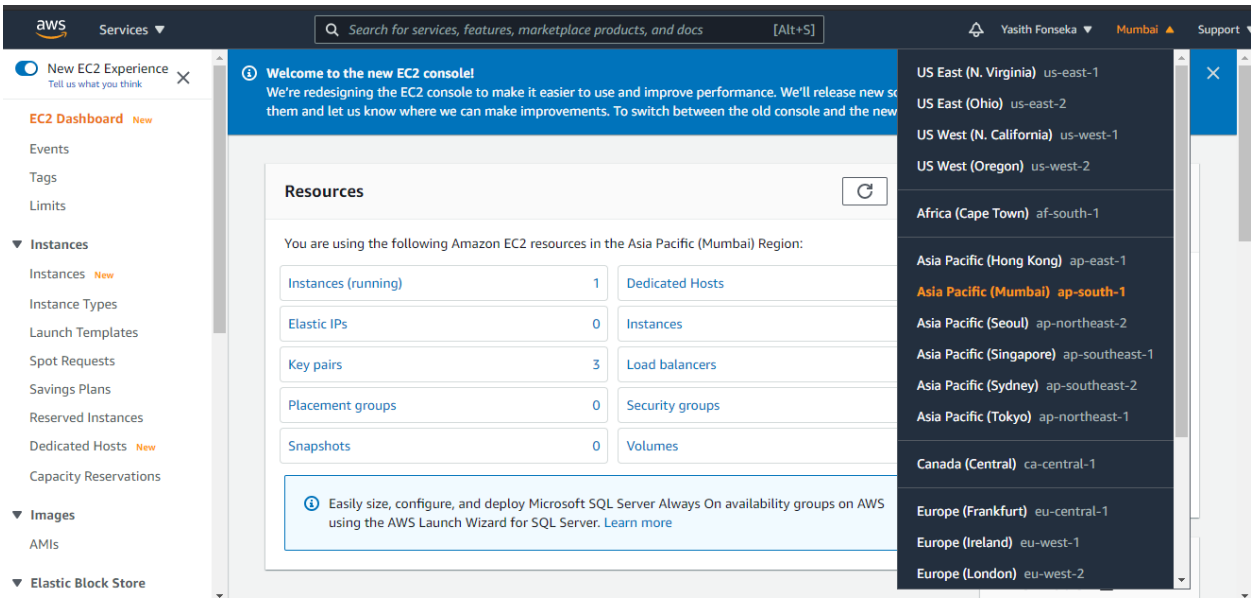
1.Sign into the console.



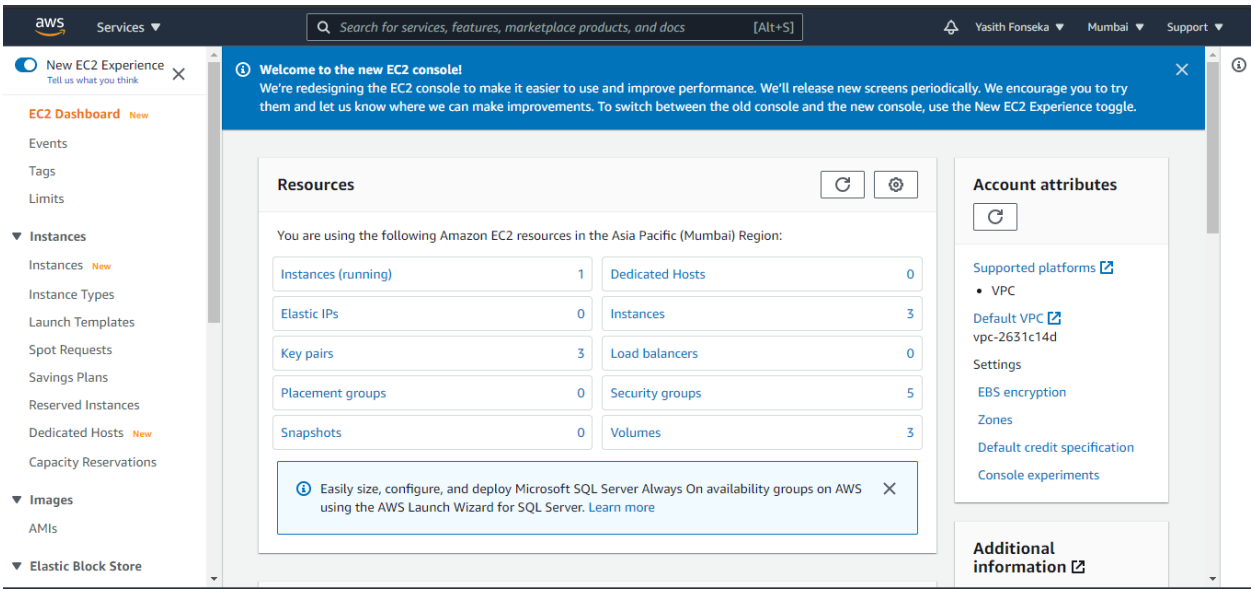
2. Select or search for EC2 instance.



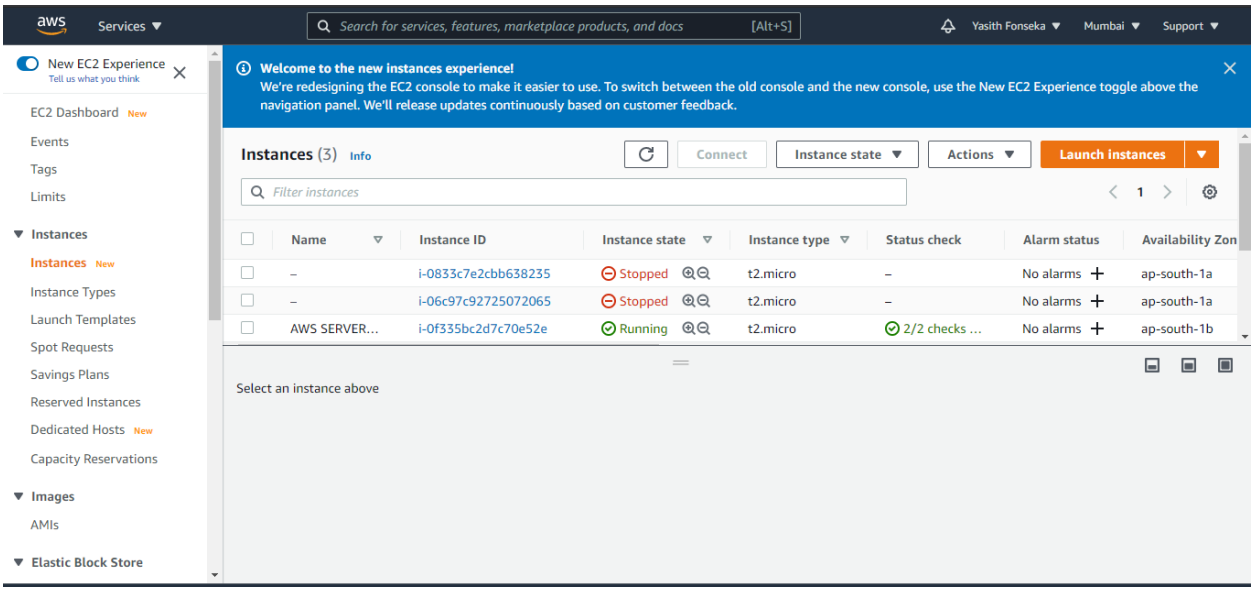
3. Select your region



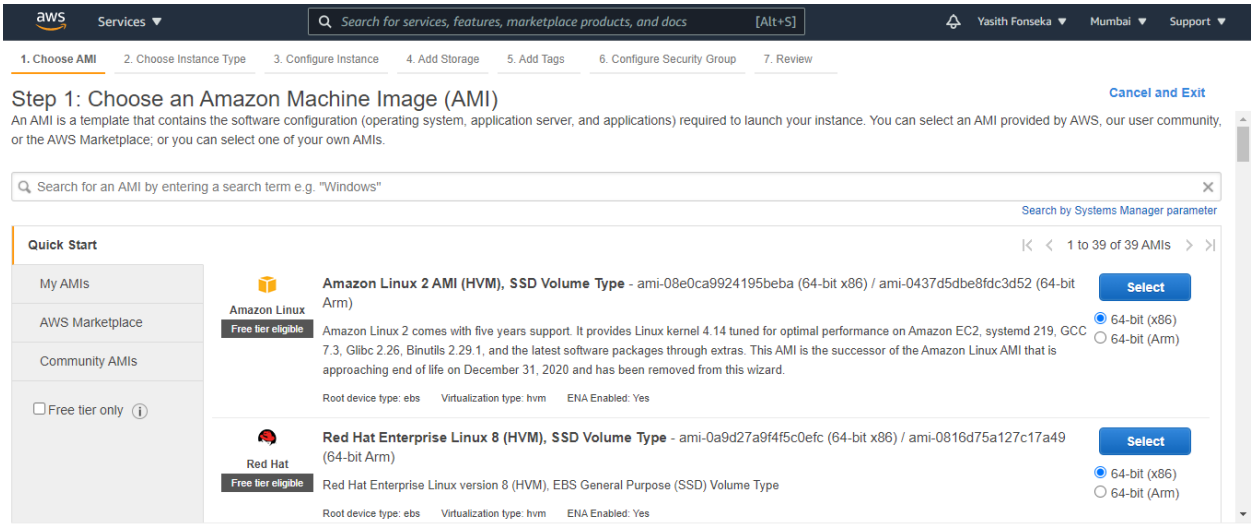
4. Select “Instances” tab



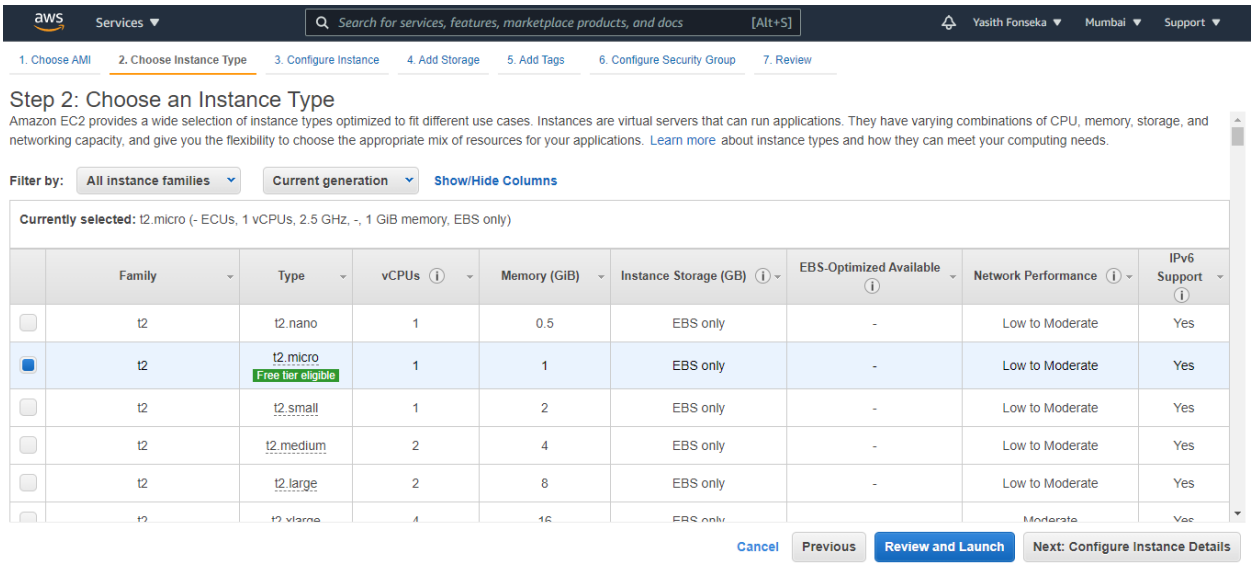
5. Select “LAUNCH INSTANCES”



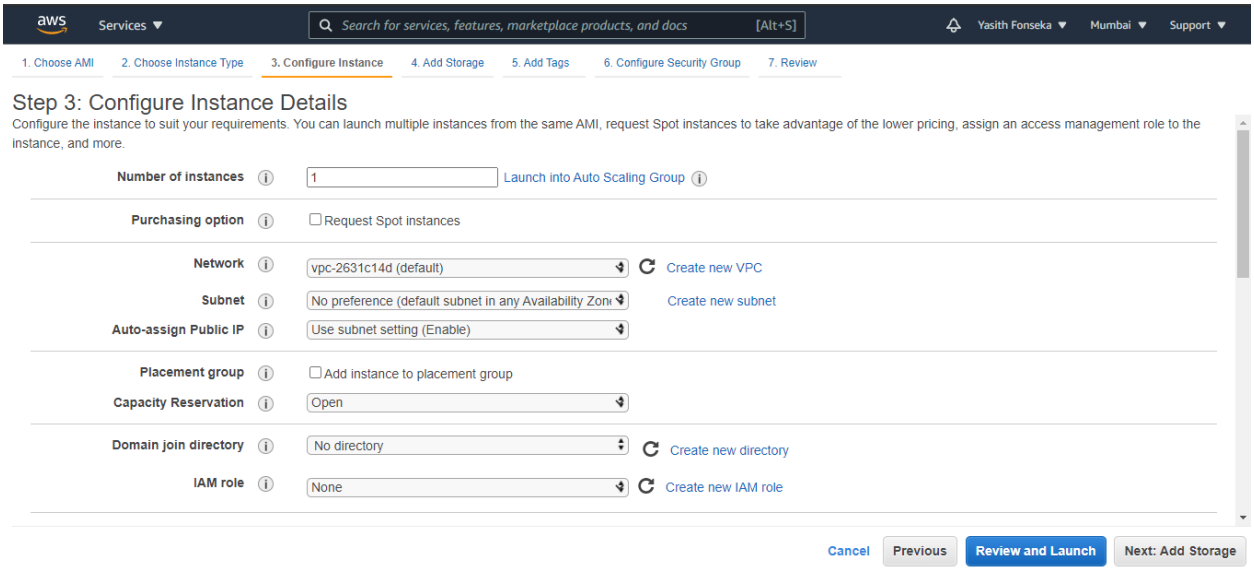
6. Choose “**Amazon Linux 2 AMI (HVM)**”



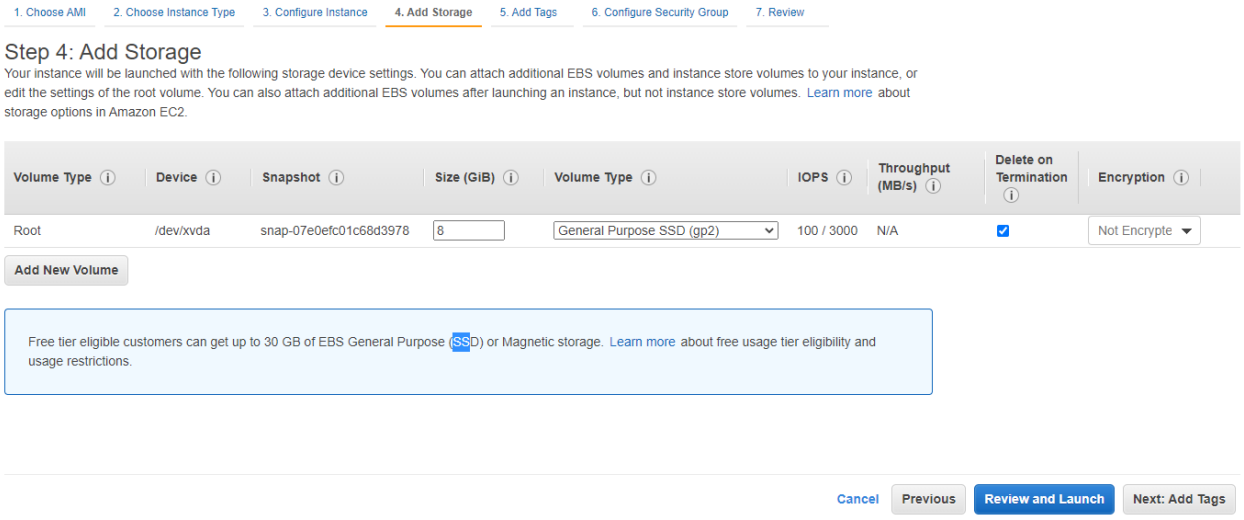
7. Selected “Free Tier Eligible” one and clicked “Configure Instance Details”.



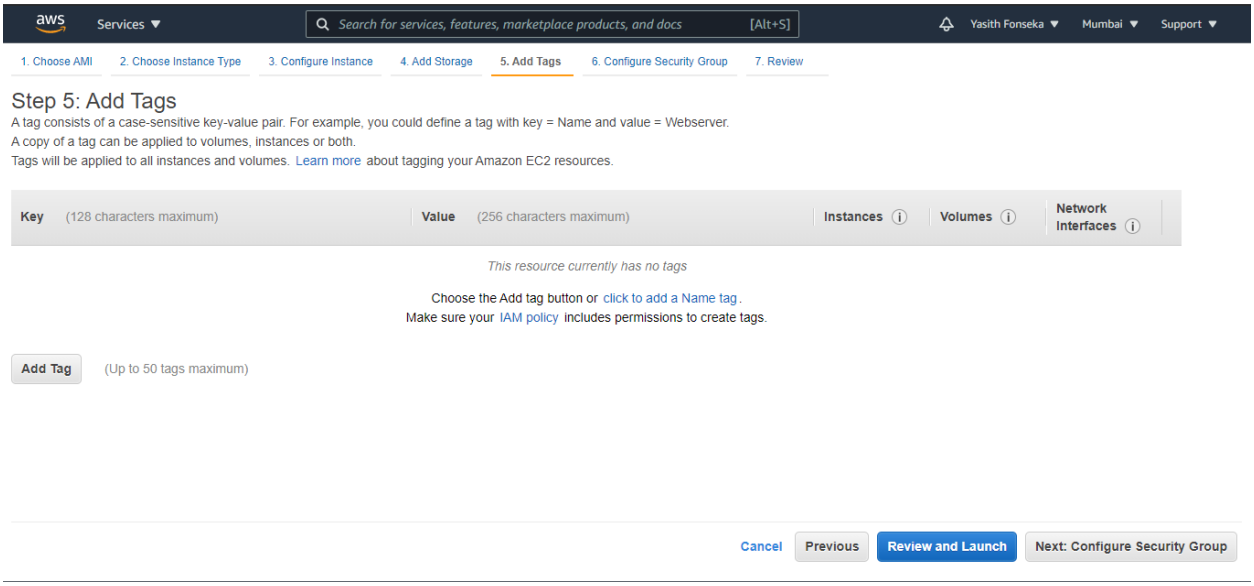
8. Kept this as it is and clicked “Add Storge”



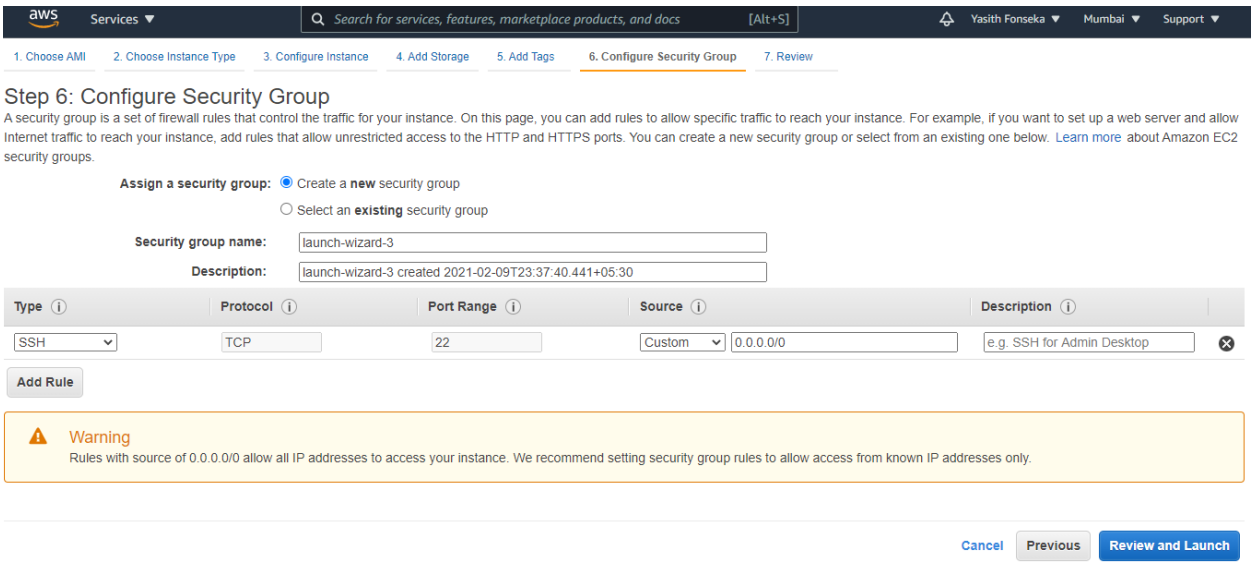
9. This one also kept as it is and clicked “Add Tags”

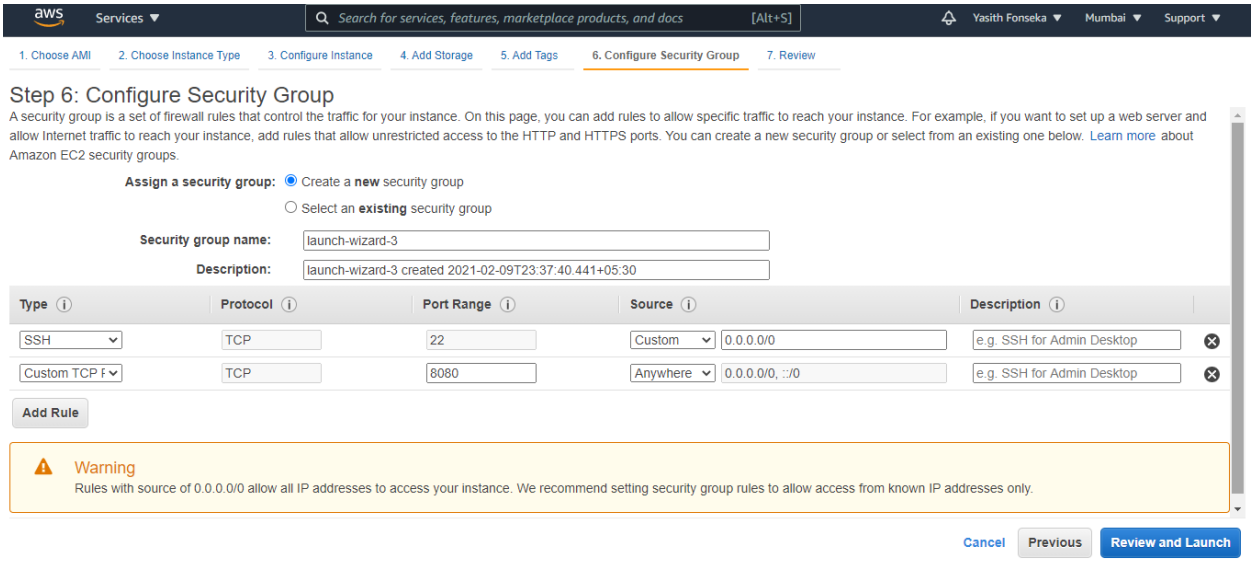


10. Kept as it is and clicked “Configure Security Groups”

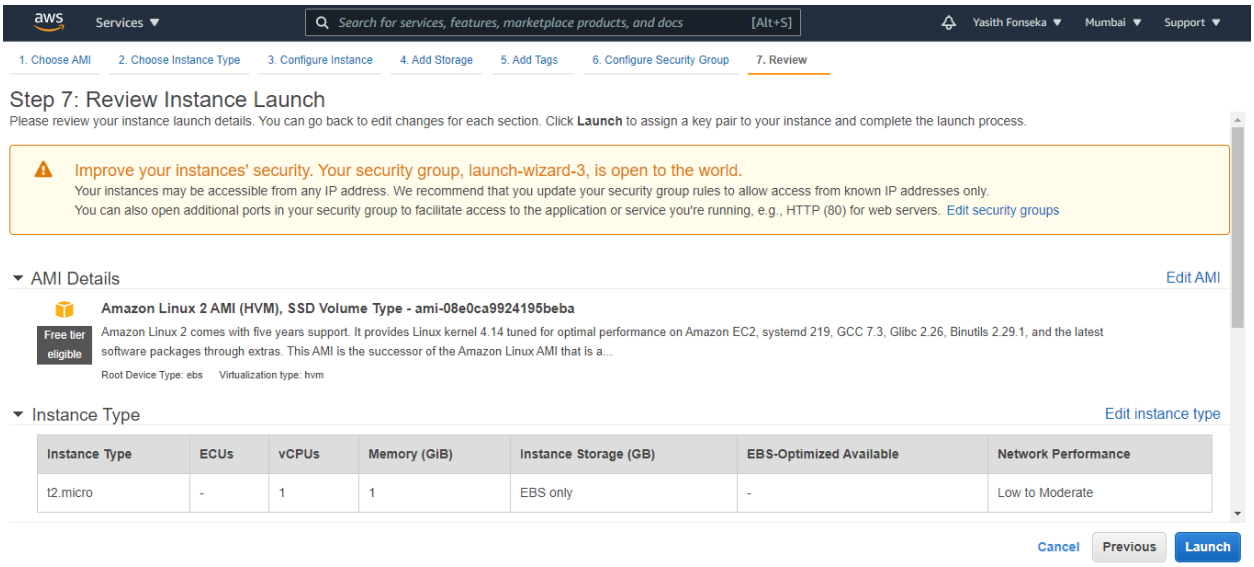


11. Here need to add a rule. For that click Add Rule button and select “Custom TCP rule and fill as shown below.

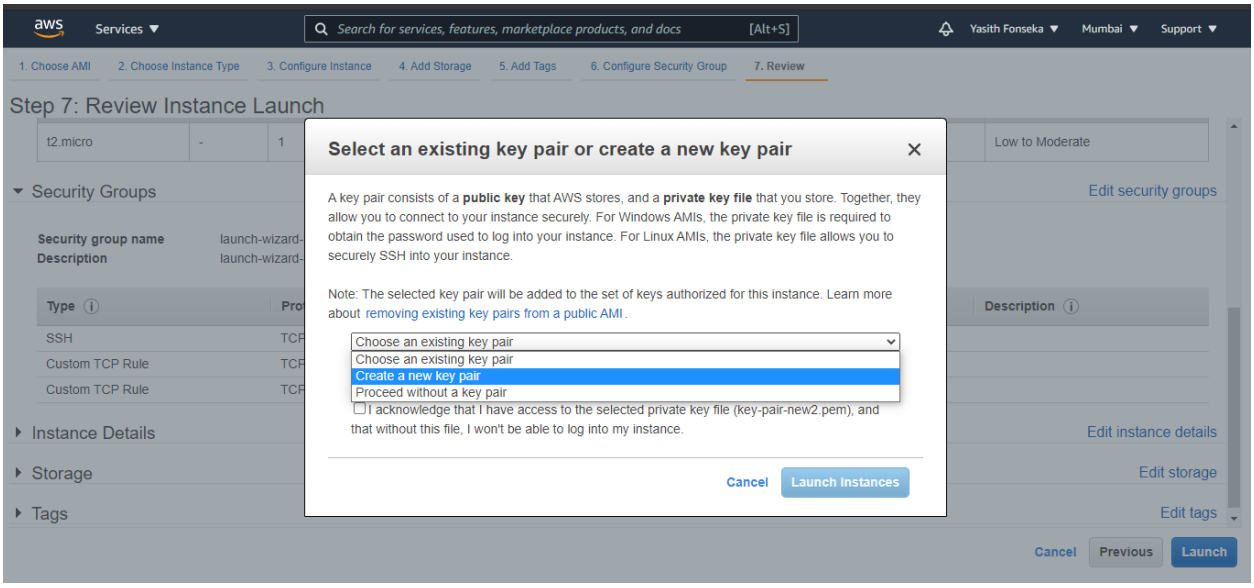




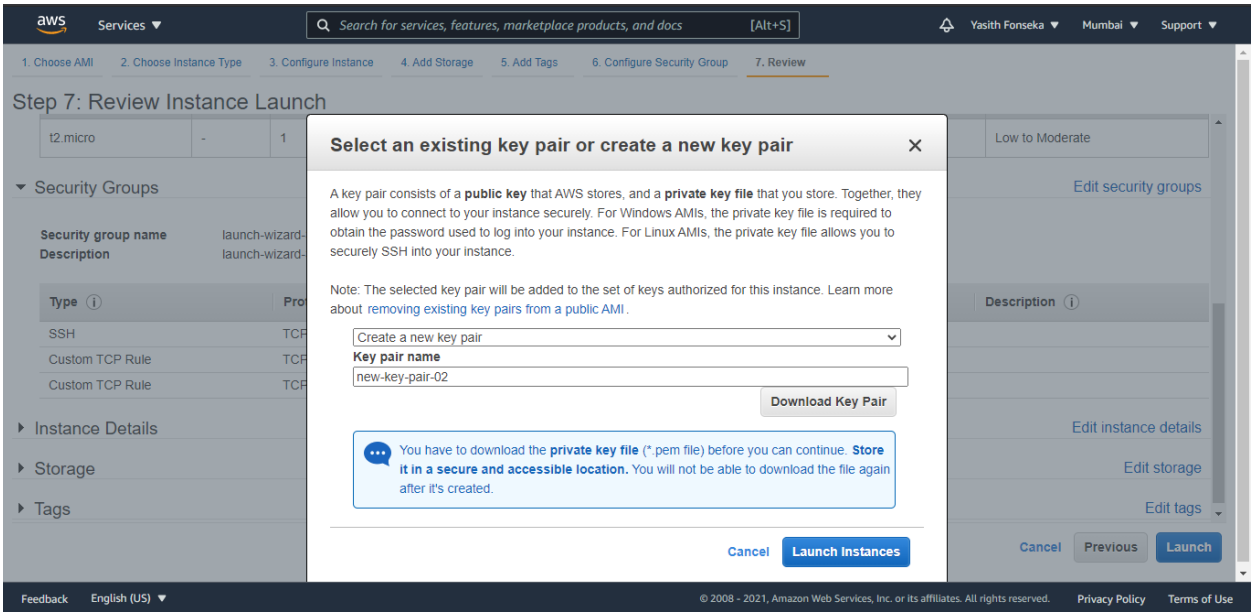
12. Keep it as it is and click “Launch”



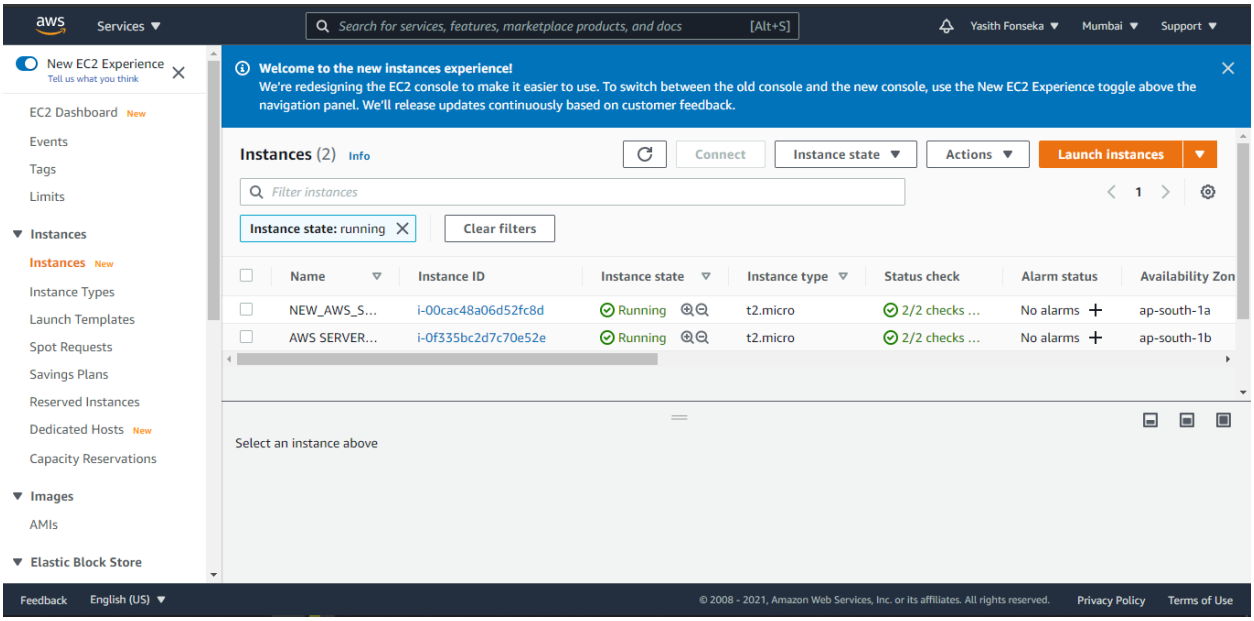
13. Here need to create a new key pair and download.



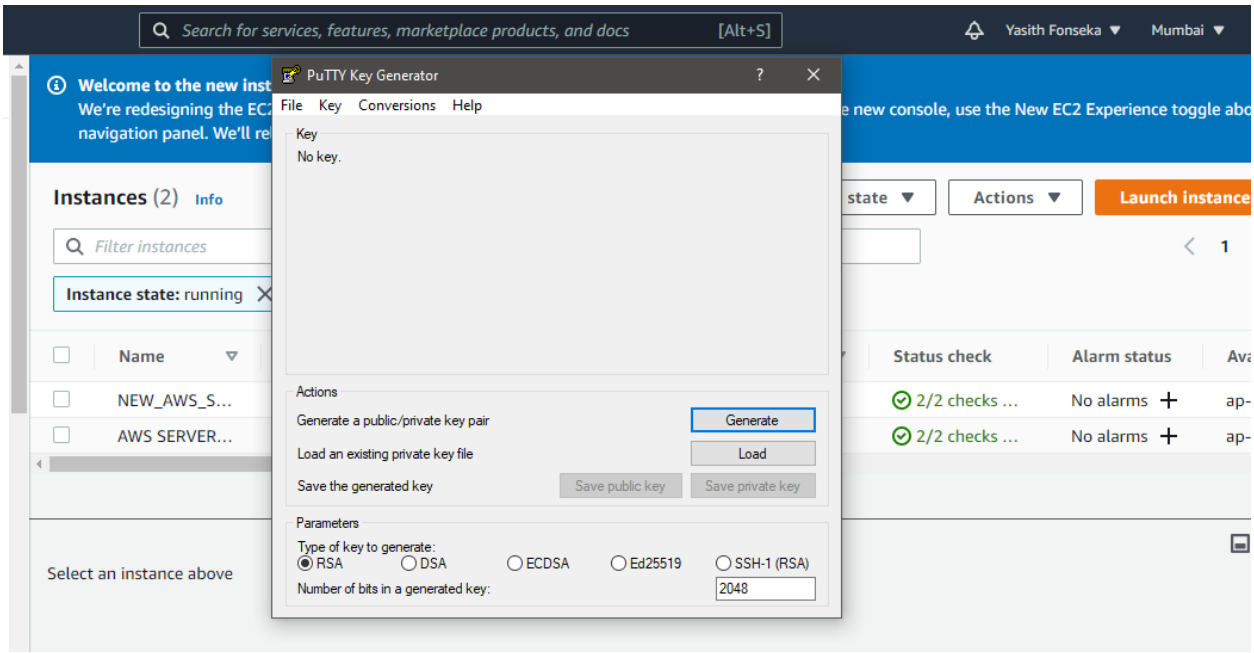
14. Click on the “Launch Instance”



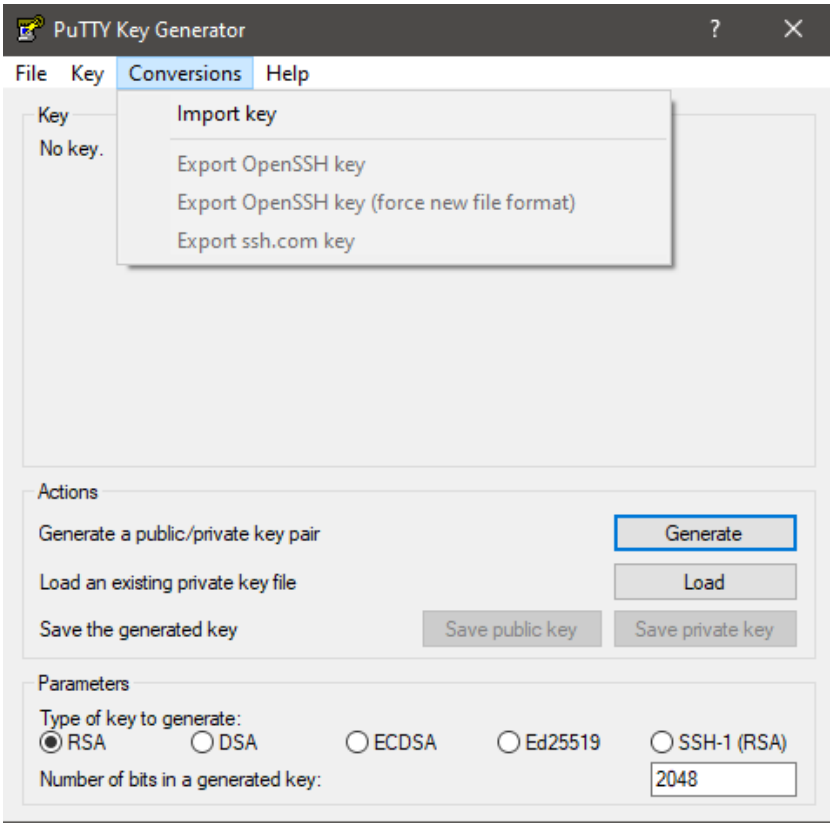
15. After few seconds ec2 instance will come to running state.



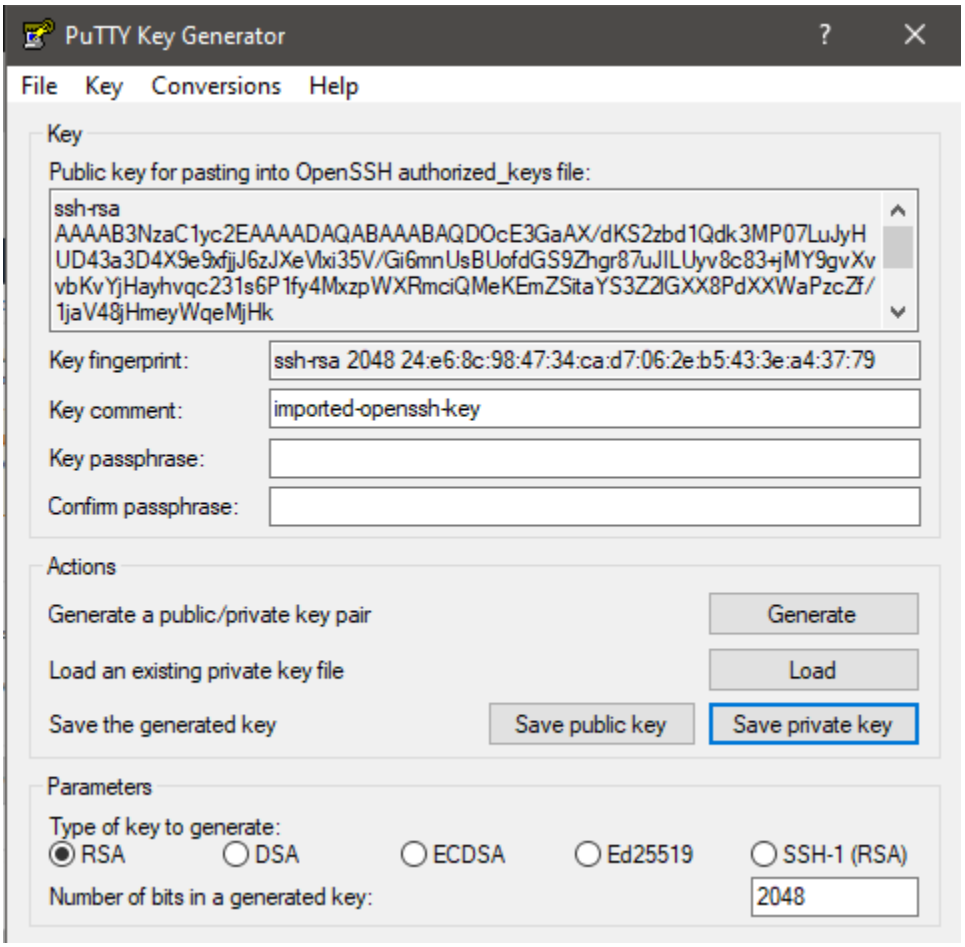
13. Then need to open PUTTYGEN



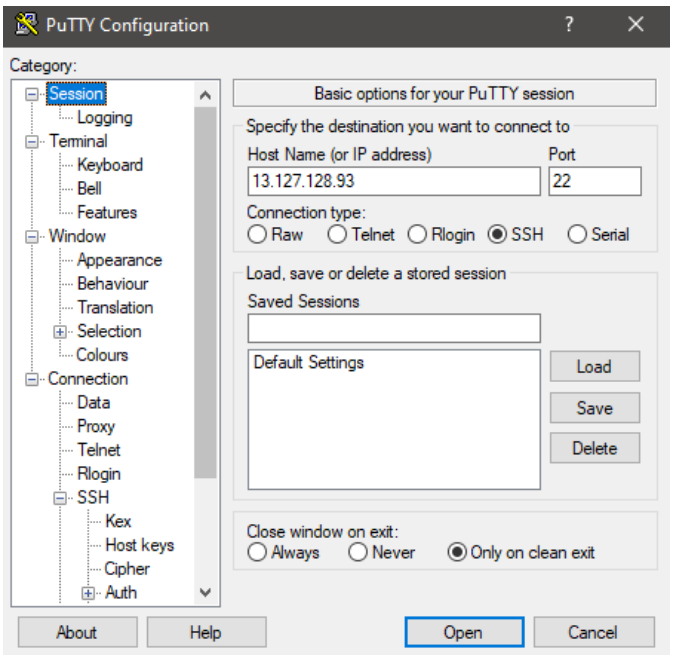
14. Here import the ".pem" key which we generated while setup setup ec2 instance.



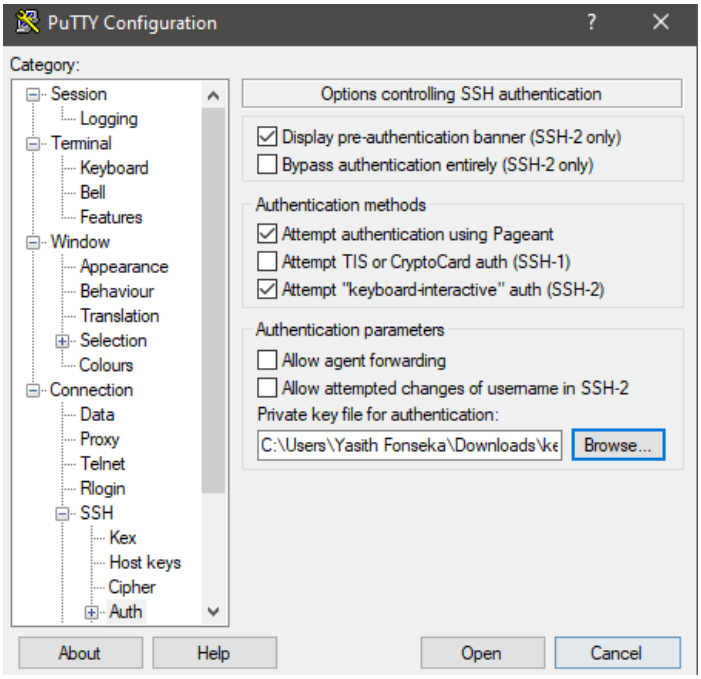
15. After importing your key click on the “Save private key” in order to save private key.



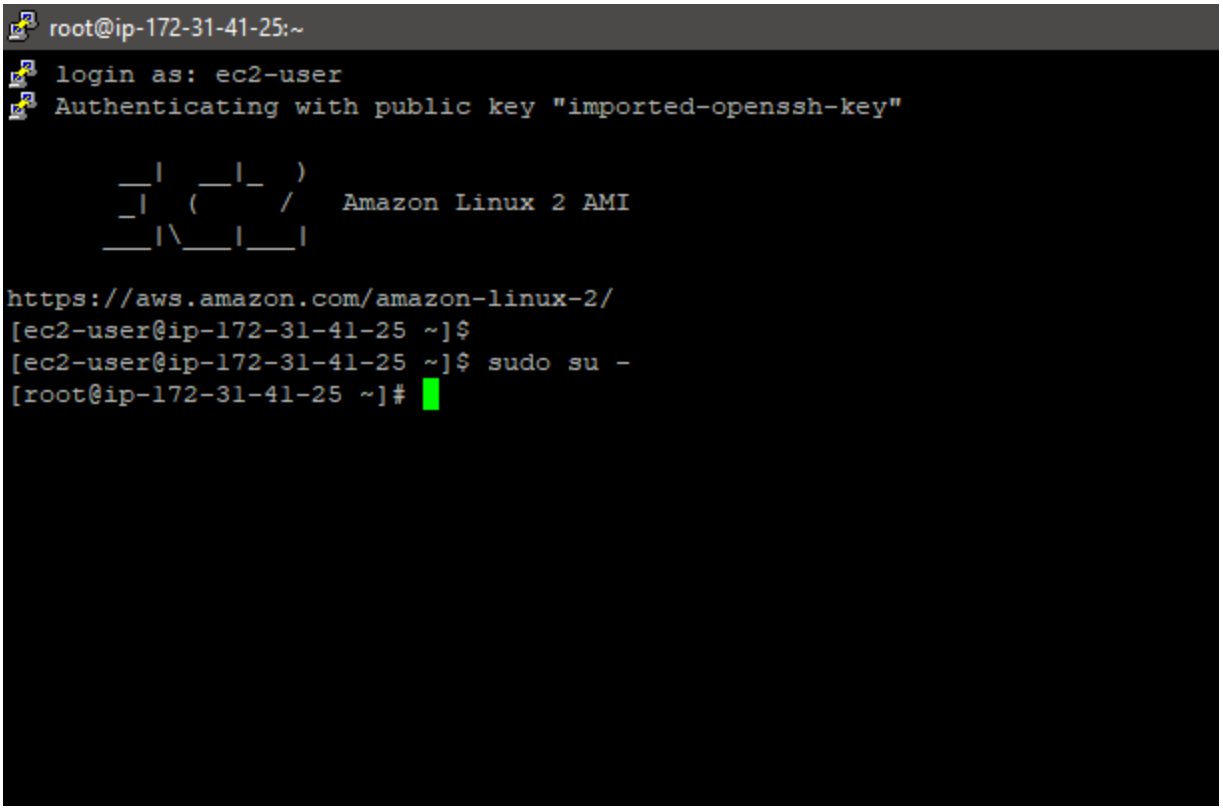
16. Then Search for “PuTTY” and open it. Give your ec2 instance’s public ipv4 address as given below.



17. Then click on “SSH” in the left side panel and click on “Auth”. Browse you saved private key in give it here. After given your private key click on “OPEN” button.

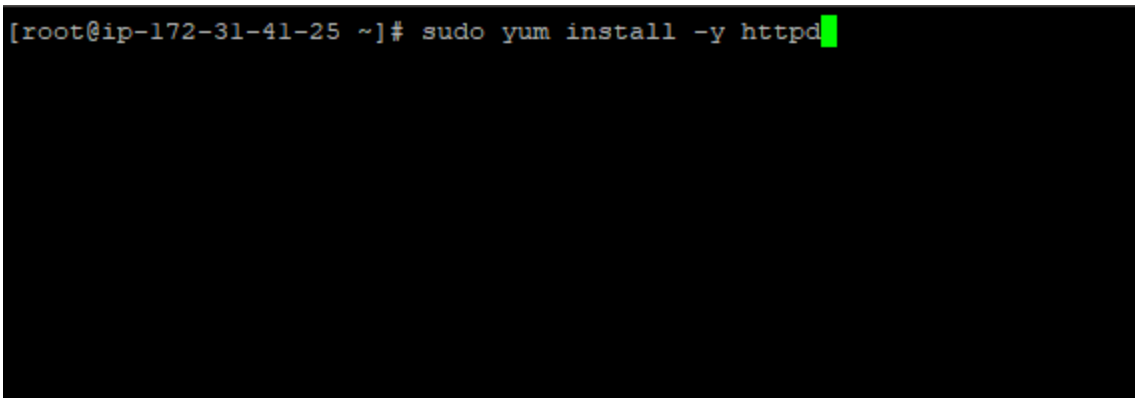


18. You will get this kind terminal. Here type “ec2-user” and hit enter. Go to the root.



19. In order to install apache do as below.

Reference(https://docs.aws.amazon.com/AmazonRDS/latest/AuroraUserGuide/CHAP_Tutorials.WebServerDB.CreateWebServer.html)

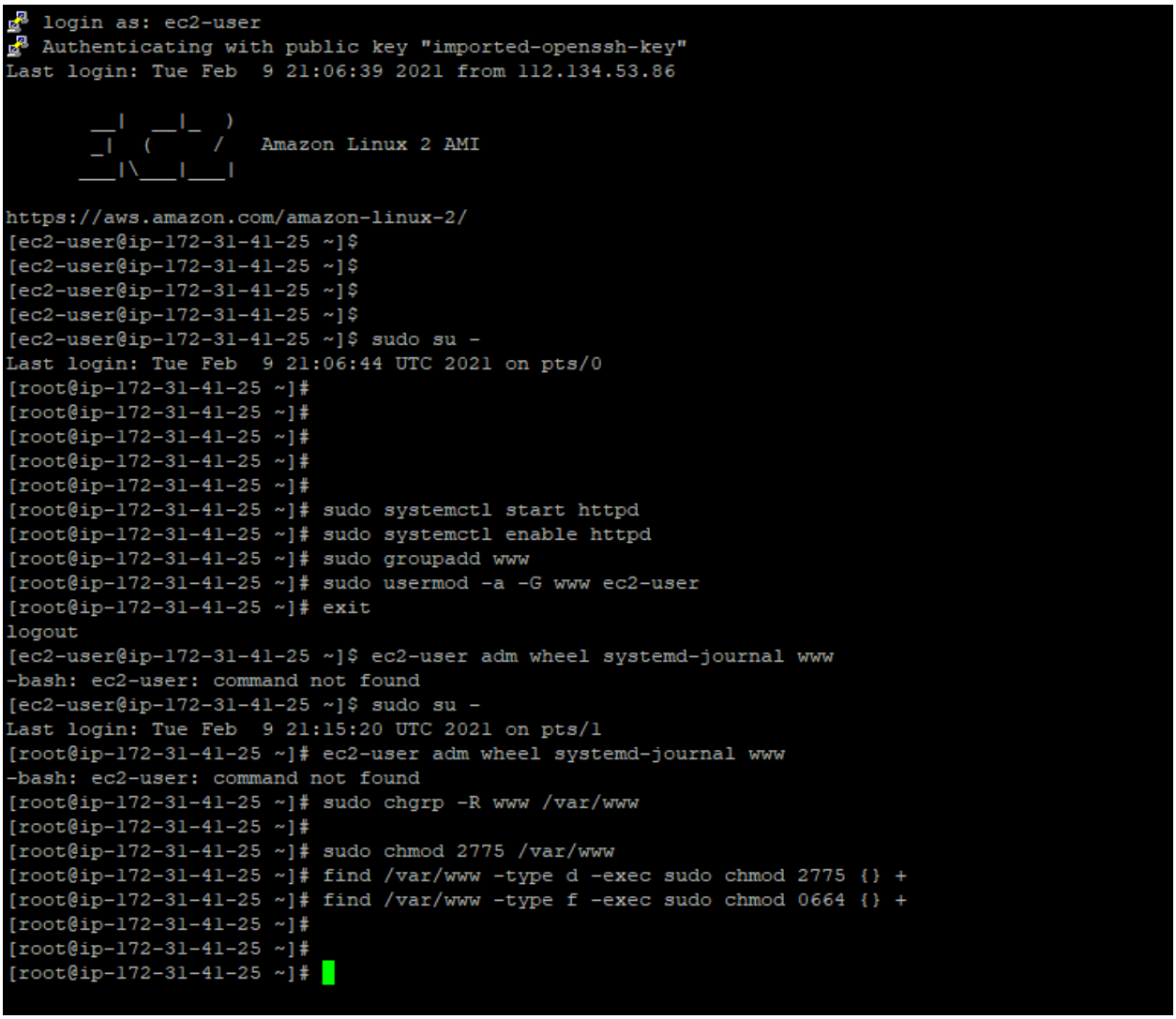


```
[root@ip-172-31-41-25 ~]# sudo yum install -y httpd
```

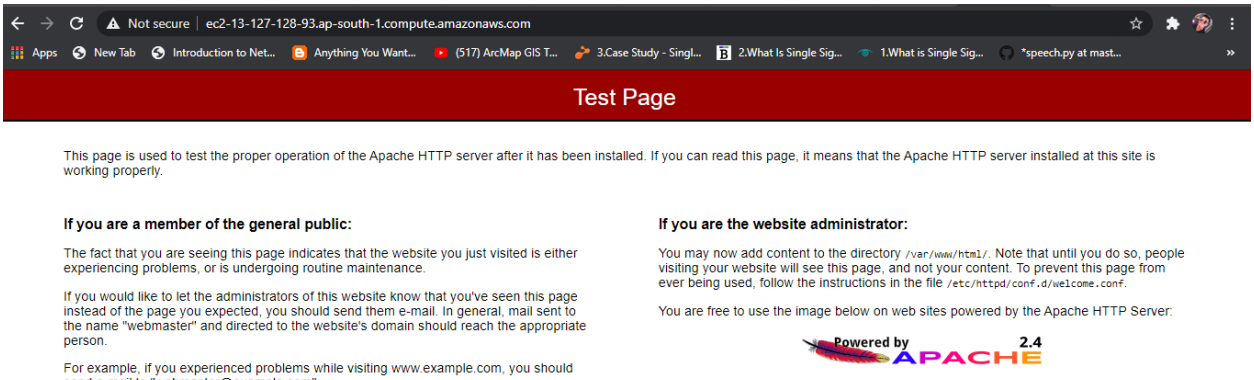
20. You can start apache service using below command.

```
[root@ip-172-31-41-25 ~]#
[root@ip-172-31-41-25 ~]#
[root@ip-172-31-41-25 ~]#
[root@ip-172-31-41-25 ~]#
[root@ip-172-31-41-25 ~]# sudo systemctl start httpd
```

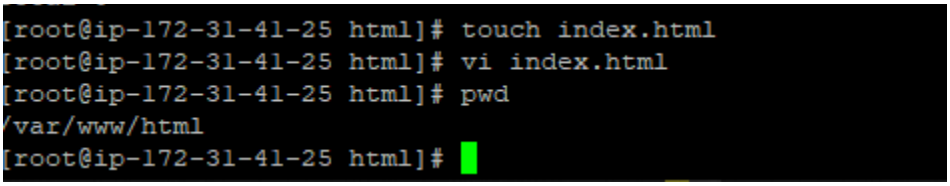
21. For further configuration need to do as follows.



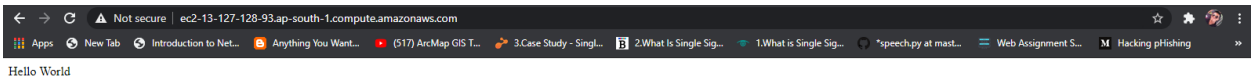
22. After done all of this as mentioned go to the dashboard and copy your "Public IPv4 DNS" and paste it as below. Then you will get page like this. Which mean you have successfully installed Apache server on you server.



23. Go to /var/www/html. Create your own page like below.



24. Then reload your page again and you will get this kind a page.



Here is the URL for public server:

(ec2-13-126-4-143.ap-south-1.compute.amazonaws.com)

Question 3

3.1

```
#!/bin/bash
if wget --spider -S "ec2-13-126-4-143.ap-south-1.compute.amazonaws.com" 2>&1 | grep -w "200\|301" ; then
    echo "Server is up"
else
    echo "Server is down. Wait till server is starting..."
    sudo systemctl start httpd
    sudo systemctl enable httpd
fi

~
~
~
~
~
~
```

3.2 – 3.4

```
#!/bin/bash

DATETIME=`date +%b-%d-%y`

webserv="http://ec2-13-126-4-143.ap-south-1.compute.amazonaws.com/"

Keyword="Hello World" # enter the keyword for test content

if curl -s "$webserv" | grep "$keyword"
then
    # if the keyword is in the conent
    echo " the website is working fine"| curl --head "$webserv" > $DATETIME.txt
    # upload txt to s3
    aws s3 cp $DATETIME.txt s3://bucket-yasith/
else
    echo "Error" > error.txt
fi
~
```

Question 4

4.1 – 4.4

```
#!/bin/bash
BACKUPTIME=`date +%b-%d-%y`
if tar -zcvf backup-$BACKUPTIME.tar.gz /var/log/*
then
        aws s3 cp backup-$BACKUPTIME.tar.gz s3://bucket-yasith/
        echo "----Backup is successfil-----"
        #echo mail -s "$BACKUPTIME Backup is successfil" beastfighter@gmail.com

else

        rm -rf backup-$BACKUPTIME.tar.gz
        echo mail -s "backup-$BACKUPTIME is not successful. Please check..." beastfighter@gmail.com
fi
~
~
~
~
~
```

4.1 Setup to take one compress file daily.

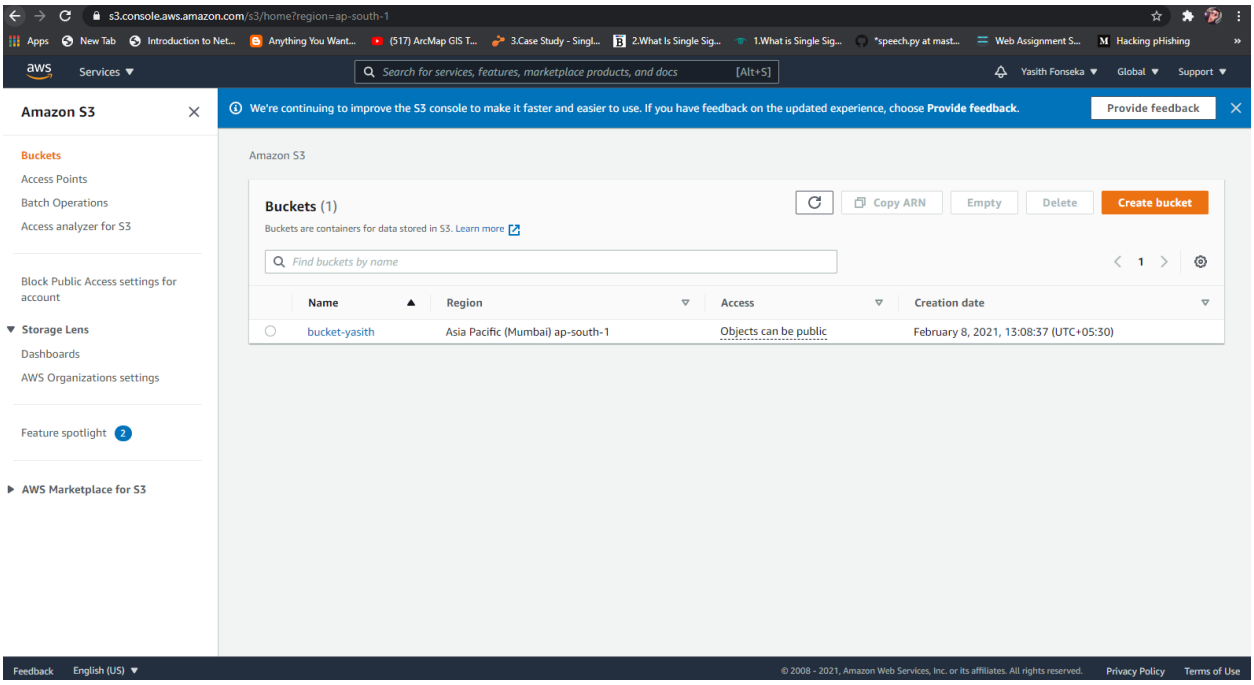
```
root@ip-172-31-2-67 ~]# crontab -e
crontab: installing new crontab
root@ip-172-31-2-67 ~]#
```

```
root@ip-172-31-2-67:~
29 0 * * * /root/backup_script.sh
~
~
~
~
~
~
~
~
~
~
~
~
~
```

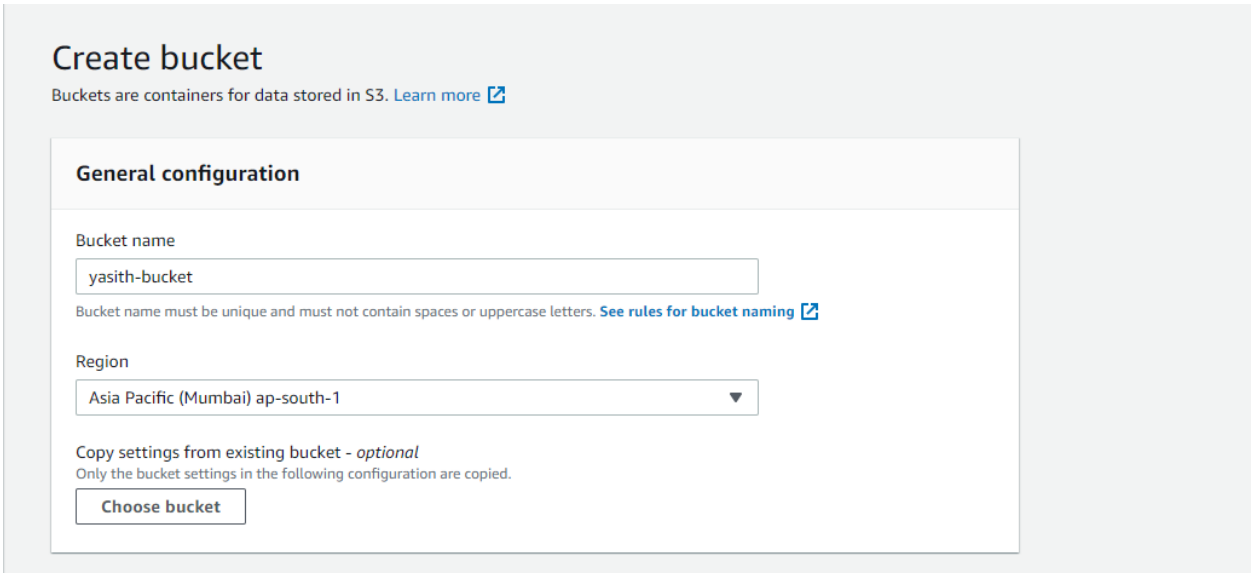
Git link for scripts -: https://github.com/yasithfonseka/assignment.git

Setup s3 bucket.

1. First need to search s3 and load the s3 dashboard
2. You will get dashboard like below and click on the “Create Bucket”.

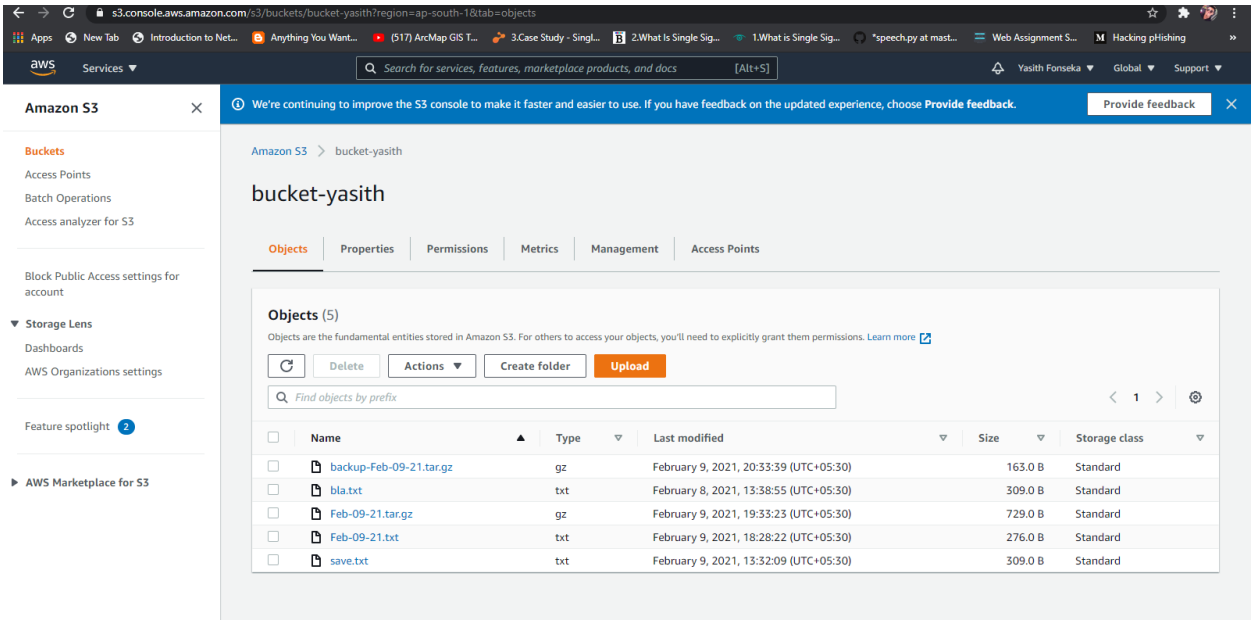


3. Here gives a unique name for bucket and click on “Create Bucket”



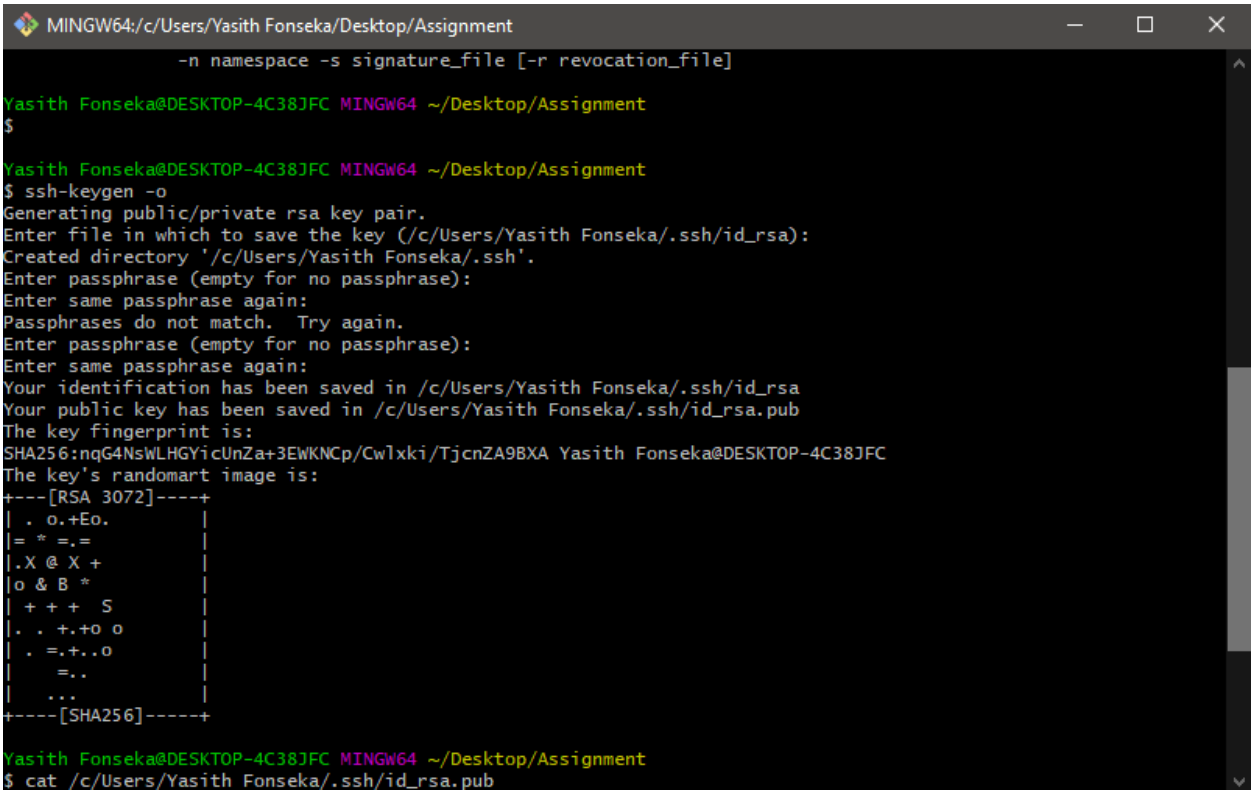
4. Then you can upload your files/ backups to the s3 bucket. In my case I used “aws s3 cp $BACKUPTIME.tar.gz s3://bucket-yasith/” command to upload files. But before that you need to configure S3 bucket in your terminal.

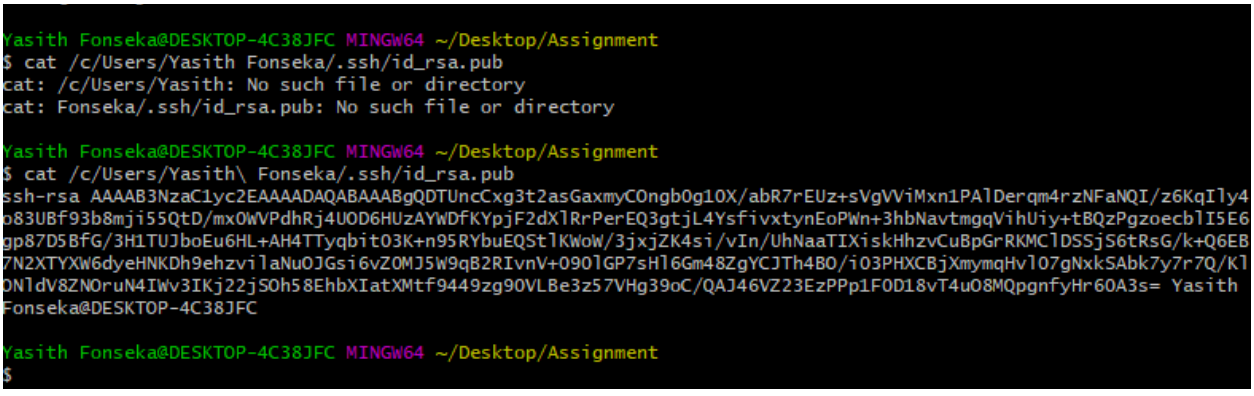
5. After that will be able to upload files like below

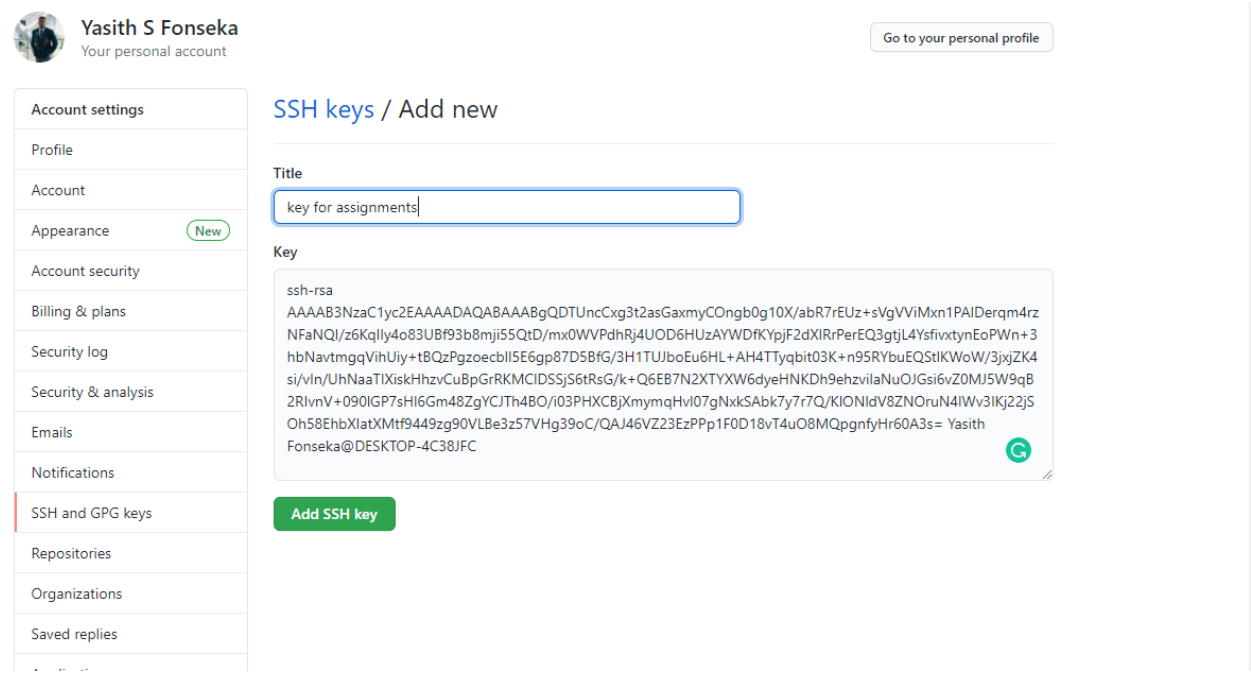


Setup git for upload files

1. Here I have attached screenshots that how I proceeded with that task.

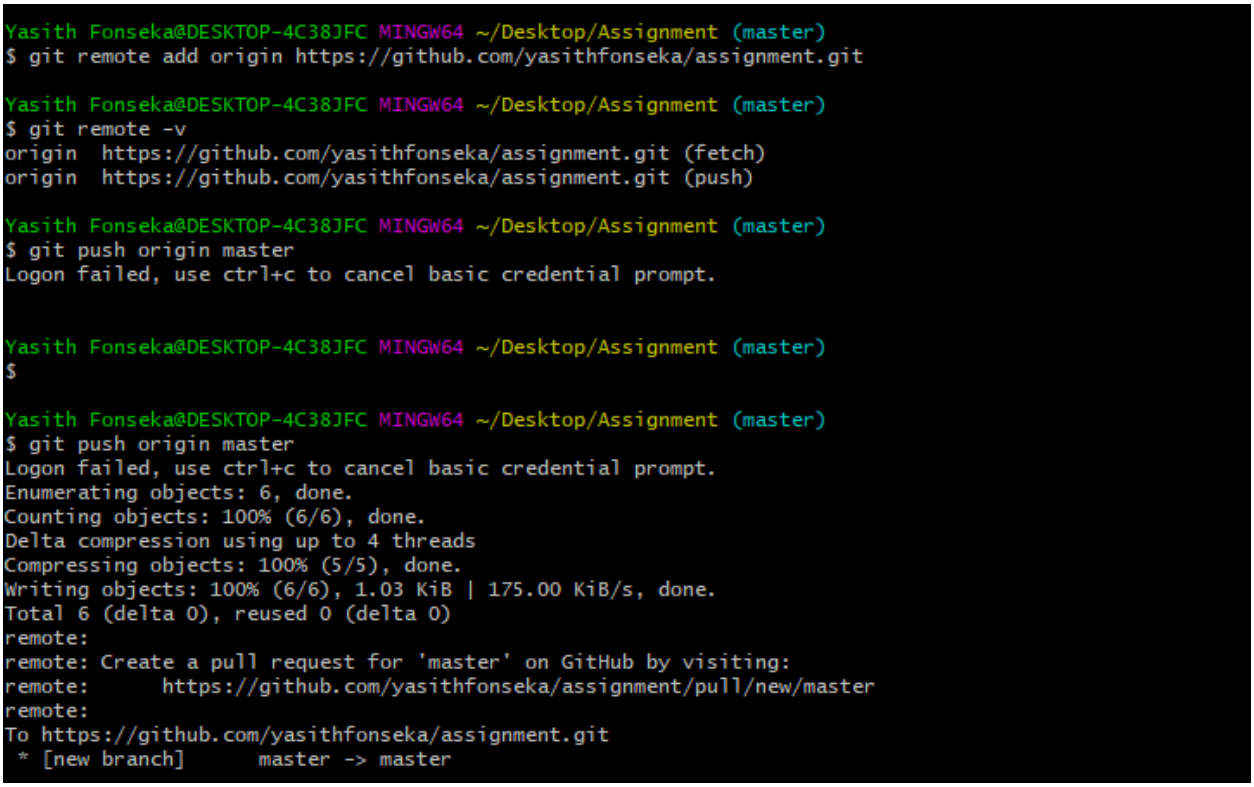







```
Yasith Fonseka@DESKTOP-4C38JFC MINGW64 ~/Desktop/Assignment
$ git init
Initialized empty Git repository in C:/Users/Yasith Fonseka/Desktop/Assignment/.git/

Yasith Fonseka@DESKTOP-4C38JFC MINGW64 ~/Desktop/Assignment (master)
$ echo "# assignmnet" >> README.md

Yasith Fonseka@DESKTOP-4C38JFC MINGW64 ~/Desktop/Assignment (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        README.md
        backup_script.sh
        check.sh
        response.sh

nothing added to commit but untracked files present (use "git add" to track)

Yasith Fonseka@DESKTOP-4C38JFC MINGW64 ~/Desktop/Assignment (master)
$ git add README.md
warning: LF will be replaced by CRLF in README.md.
The file will have its original line endings in your working directory

Yasith Fonseka@DESKTOP-4C38JFC MINGW64 ~/Desktop/Assignment (master)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   README.md

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        backup_script.sh
        check.sh
        response.sh


Yasith Fonseka@DESKTOP-4C38JFC MINGW64 ~/Desktop/Assignment (master)
$ git add .
```

```
Yasith Fonseka@DESKTOP-4C38JFC MINGW64 ~/Desktop/Assignment (master)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   README.md
        new file:   backup_script.sh
        new file:   check.sh
        new file:   response.sh


Yasith Fonseka@DESKTOP-4C38JFC MINGW64 ~/Desktop/Assignment (master)
$ git commit -m "first commit"

*** Please tell me who you are.

Run

  git config --global user.email "you@example.com"
  git config --global user.name "Your Name"

to set your account's default identity.
Omit --global to set the identity only in this repository.

fatal: unable to auto-detect email address (got 'Yasith Fonseka@DESKTOP-4C38JFC.(none)')

Yasith Fonseka@DESKTOP-4C38JFC MINGW64 ~/Desktop/Assignment (master)
$ git config --global user.email "yasithfonseka123@gmail.com"

Yasith Fonseka@DESKTOP-4C38JFC MINGW64 ~/Desktop/Assignment (master)
$ git config --global user.name "Yasith Fonseka"

Yasith Fonseka@DESKTOP-4C38JFC MINGW64 ~/Desktop/Assignment (master)
$ git commit -m "first commit"
[master (root-commit) d443e3b] first commit
 4 files changed, 39 insertions(+)
 create mode 100644 README.md
 create mode 100644 backup_script.sh
 create mode 100644 check.sh
 create mode 100644 response.sh
```



https://github.com/yasithfonseka/assignment/compare/master?expand=1